



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|-----------------|-------------|----------------------|---------------------|------------------|
| 09/760,321      | 01/12/2001  | Markus Meyer         | P-4355              | 2684             |

7590 04/06/2005

Forrest Gunnison  
Gunnison, McKay & Hodgson, L.L.P.  
Suite 220  
1900 Garden Road  
Monterey, CA 93940

|          |
|----------|
| EXAMINER |
|----------|

BULLOCK JR, LEWIS ALEXANDER

|          |              |
|----------|--------------|
| ART UNIT | PAPER NUMBER |
|----------|--------------|

2195

DATE MAILED: 04/06/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

## Office Action Summary

**Application No.**

09/760,321

**Applicant(s)**

MEYER, MARKUS

**Examiner**

Lewis A. Bullock, Jr.

**Art Unit**

2195

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

### Status

- 1) ☒ Responsive to communication(s) filed on 15 November 2004.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

### Disposition of Claims

- 4) ☒ Claim(s) 4, 6, 11-14, 18, 19 and 21-28 is/are pending in the application.
- 4a) Of the above claim(s) 23-28 is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 4, 6, 11-14, 18, 19, 21 and 22 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

### Priority under 35 U.S.C. § 119

- 12) ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☒ All b) ☐ Some \* c) ☐ None of:
1. ☒ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

### Attachment(s)

- |   |   |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)   | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)  | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date <u>1/3/05</u> . | 6) <input type="checkbox"/> Other: _____  |

**DETAILED ACTION**

***Election/Restrictions***

1. Applicant's election of Group I, claims 1-22 in the reply filed on 11/15/04 is acknowledged. Because applicant did not distinctly and specifically point out the supposed errors in the restriction requirement, the election has been treated as an election without traverse (MPEP § 818.03(a)).
2. This application contains claims 23-28 are drawn to an invention nonelected without traverse in Paper No. 11/15/04. A complete reply to the final rejection must include cancellation of nonelected claims or other appropriate action (37 CFR 1.144) See MPEP § 821.01.

***Claim Rejections - 35 USC § 102***

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. Claims 6, 12-14, 18, 19, 21, and 22 are rejected under 35 U.S.C. 102(b) as being anticipated by "COM/CORBA Interworking" by Digital Equipment Corporation (DEC).

As to claim 6, DEC teaches a method for using functionality in a second execution environment (CORBA System) in a first execution environment (COM / OLE System) comprising: calling a method in a proxy interface (IclassFactory::CreateInstance / pSomeInterface::function / Surrogate Server) in the first

execution environment (COM / OLE System, via the COM/OLE Client); and converting the method call (transformation of arguments before forward request) by the proxy interface to a corresponding method call (CosLifeCycle::GenericFactory::create\_object / SomeInterface::OperationName) for execution in the second execution environment (CORBA System, via invoking the CORBA object) and dispatching the method call to the second environment after the conversion wherein the converting comprises: using a type description (mapping) to convert parameters from the first execution environment to the second execution environment (pgs. 11-16, in particular pg. 14, "Once the COM client is returned the interface pointer, it can begin performing operations on the object itself. Each time the COM client calls a member function on an interface pointer, the surrogate COM object will be contacted. The implementation of a member function may need to perform any required transformation of arguments (e.g. convert strings between UNICODE and ANSI) and then forward the request to the corresponding CORBA object."; see also pgs. 17-79, in particular pgs. 27-28, Mapping for Operations).

As to claim 12, refer to claim 6 for rejection.

As to claims 13 and 14, DEC teaches dispatching the method call (request / IsomeInterface::function) for execution in the second execution environment to the second execution environment (CORBA System) by the proxy interface IclassFactory::CreateInstance / pSomeInterface::function / Surrogate Server) wherein the parameters sent / results returned are converted between execution environments

Art Unit: 2195

by a proxy interface using a type description (mapping) (transformation of arguments / results before forward) (pgs. 11-16, in particular pg. 14, "Once the COM client is returned the interface pointer, it can begin performing operations on the object itself. Each time the COM client calls a member function on an interface pointer, the surrogate COM object will be contacted. The implementation of a member function may need to perform any required transformation of arguments (e.g. convert strings between UNICODE and ANSI) and then forward the request to the corresponding CORBA object."; pg. 11, "In this role, the COM surrogate object performs any transformations necessary in order to make the associated request on a CORBA object. These transformations include mapping COM'S use of global unique identifiers (GUIDs), conversion of error information which needs to be returned, and the mapping between CORBA interface pointers and CORBA object references."; see also fig.3-4).

As to claims 18, 19, 21 and 22, reference is made to a program product that corresponds to the method of claims 12, 13 and 14 and is therefore met by the rejection of claims 12, 13 and 14 above.

### ***Claim Rejections - 35 USC § 103***

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over "COM/CORBA Interworking" by Digital Equipment Corporation (DEC).

As to claim 11, DEC teaches communication between environments wherein one environment communicates with another environment through a proxy interface that translates method calls from one environment to another. DEC teaches that the environments are COM and CORBA. However, DEC does not teach that the environments use C++ programming language. Official Notice is taken in that the COM environment has C++ constructs and that it would be obvious to one skilled in the art that the COM environment is a C++ programming language execution environment that communicates with another environment.

3. Claim 4 is rejected under 35 U.S.C. 103(a) as being unpatentable over "COM/CORBA Interworking" by DEC1 in view of COM-CORBA Interworking RFP Part A" by DEC2.

As to claim 4, DEC1 teaches a method comprising: generating a binary specification object (client written in the COM specification / Microsoft Interface Definition Language / Microsoft Object Definition Language (COM client)) for a first execution environment (COM / OLE system); generating a binary specification object (server written in the CORBA specification / CORBA Interface Definition Language (CORBA server)) for a second execution environment (CORBA system); and generating a bridge object (COM-CORBA Inter-working / surrogate server) wherein the bridge object is used in mapping objects from the second execution environment (CORBA

servers) to the first execution environment (COM system) (pg. 10, "In fact, the CORBA object is actually presented to the COM client as a COM object. It does this by wrapping the reference to the CORBA object inside of a COM object."; pg. 11, "One of the most obvious approaches is to create an executable which resides on the same platform where COM/OLE is installed...The Surrogate COM Server approach to providing interoperability between COM and CORBA uses a COM/OLE object server that acts as a surrogate for the remote CORBA object server...The surrogate server approach requires that a COM object be created for each CORBA object that will be accessible to COM/OLE client applications. The COM objects acts as a surrogate for the CORBA objects, thus the name Surrogate Server."; pg. 74, "The ability of a COM client to invoke an operation on an object in a CORBA environment where the COM client and CORBA object are independently developed. The CORBA object interfaces shall be assumed to be specified using an OMG IDL interface description The COM client shall be assumed to have only COM interface specifications available to it."); However, DEC1 does not teach the bridge object generate a proxy wrapping an interface in the second execution environment.

DEC2 teaches the bridge object (bridge) generating a proxy (view object) wrapping an interface in the second execution environment (pg. 17, "On the left we have a client in object system A, which wants to send a request to a target object in system B, seen on the right. We refer to the entire (conceptual entity) that provides the mapping as a bridge. The goal is to map and deliver any request from the client transparently to the target. To do so, we first provide an object in system A called a View. The View is an

object in system A which presents the identity and interface of the target in system B mapped to the vernacular of system A, and is described as an A View of a B target. The View exposes an interface, called the View Interface, which is isomorphic to the target's interface in System B. The methods of the View Interface convert requests from System A clients into requests on the target's interface in System B. The View is a component of the bridge." ). Therefore, since the View is a component of the bridge, the view is created when the bridge is created. Therefore, it would be obvious to one skilled in the art to combine the teachings of DEC1 with the teachings of DEC2 in order to facilitate the inter-working of object systems

### ***Response to Arguments***

7. Applicant's arguments filed 11/15/04 have been fully considered but they are not persuasive. Applicant argued that the cited teachings of DEC does not teach the specific way to convert parameters. The examiner disagrees. The cited limitation discloses using a type description to convert parameters from the first execution environment to the second execution environment. A type description as interpreted in the claim usage is a mapping from one execution environment to another execution environment. The examiner as referred to one illustration of the transformation wherein there exists mappings of COM's use of global unique identifiers (GUIDs), conversion of error information which needs to be returned, and the mapping between CORBA interface pointers and CORBA object references (pg. 11). The document also illustrates on pages 17-79, the mapping of data types and/or data structures from one



environment to another environment. Therefore, the cited reference illustrates multiple uses of a type description (mapping) for converting parameters from one environment to another environment.

Applicant then argues obvious rejection to claim 11 is not well founded since (a) the second execution environment is the CORBA system and not the COM system, and (b) the additional information does not overcome the deficiency of the primary reference. The examiner disagrees. As explained above, DEC adequately teaches that a specific way to convert parameters by using a type description (mapping) to convert parameters from the first execution environment to the second execution environment. Applicant is referred to the various other pages 17-79 of the cited reference for a more detailed showing of how parameters are mapped to/from the environments. Secondly, the second environment would also qualify as a COM environment. Previously cited pages 14-16, illustrate the client is the CORBA system and the invoked second system, is the COM system. In addition, a further reading of the cited reference also illustrates that CORBA data types, OMG IDL types, are the equivalent of C++ class definitions (pg. 17). Therefore, since these types are executed on the CORBA environment, the CORBA environment is a C++ programming language execution environment. In addition, the examiner has a couple of other references that are cited in the Notice of References Cited that teach that CORBA server objects are implemented in any programming language, including C++, since CORBA provides multiple programming language mappings for OMG IDL (see "C++ Programming with CORBA reference; and Understanding CORBA reference). Therefore, the CORBA

Art Unit: 2195

environment executes C++ programming code and is a C++ programming language execution environment.

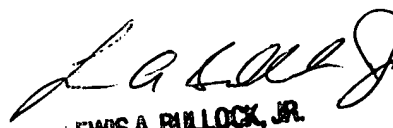
### ***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Lewis A. Bullock, Jr. whose telephone number is (571) 272-3759. The examiner can normally be reached on Monday-Friday, 8:30 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

April 1, 2005

  
LEWIS A. BULLOCK, JR.  
PRIMARY EXAMINER